Enhancing Deformable Local Features by Jointly Learning to Detect and Describe Keypoints

Guilherme Potje[1] Felipe Cadar[1] André Araujo[2]
Renato Martins[3,4] Erickson R. Nascimento[1,5]

[1]Universidade Federal de Minas Gerais [2]Google Research
[3]Université de Bourgogne [4]Université de Lorraine, LORIA, Inria [5]Microsoft

{guipotje,cadar,erickson}@dcc.ufmg.br, renato.martins@u-bourgogne.fr, andrearaujo@google.com

Abstract

Local feature extraction is a standard approach in computer vision for tackling important tasks such as image matching and retrieval. The core assumption of most methods is that images undergo affine transformations, disregarding more complicated effects such as non-rigid deformations. Furthermore, incipient works tailored for non-rigid correspondence still rely on keypoint detectors designed for rigid transformations, hindering performance due to the limitations of the detector. We propose DALF (Deformation-Aware Local Features), a novel deformation-aware network for jointly detecting and describing keypoints, to handle the challenging problem of matching deformable surfaces. All network components work cooperatively through a feature fusion approach that enforces the descriptors' distinctiveness and invariance. Experiments using real deforming objects showcase the superiority of our method, where it delivers 8% improvement in matching scores compared to the previous best results. Our approach also enhances the performance of two real-world applications: deformable object retrieval and non-rigid 3D surface registration. Code for training, inference, and applications are publicly available at verlab.dcc.ufmg.br/descriptors/dalf_cvpr23.

1. Introduction

Finding pixel-wise correspondences between images depicting the same surface is a long-standing problem in computer vision. Besides varying illumination, viewpoint, and distance to the object of interest, real-world scenes impose additional challenges. The vast majority of the correspondence algorithms in the literature assume that our world is rigid, but this assumption is far from the truth. It is noticeable that the community invests significant efforts into novel architectures and training strategies to improve image matching for rigid scenes [6, 19, 26, 34, 37, 42], but disregards the fact that many objects in the real world can deform in more complex ways than an affine transformation.

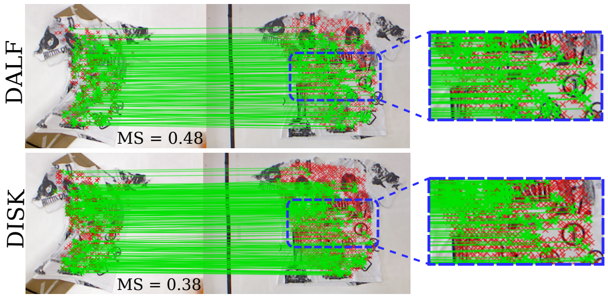

Figure 1. **Image matching under deformations**. We propose DALF, a deformation-aware keypoint detector and descriptor for matching deformable surfaces. DALF (top) enables local feature matching across deformable scenes with improved matching scores (MS) compared to state-of-the-art, as illustrated with DISK [37]. Green lines show correct matches, and red markers, the mismatches.

Many applications in industry, medicine, and agriculture require tracking, retrieval, and monitoring of arbitrary deformable objects and surfaces, where a general-purpose matching algorithm is needed to achieve accurate results. Since the performance of standard affine local features significantly decreases for scenarios such as strong illumination changes and deformations, a few works considering a wider class of transformations have been proposed [24, 25, 30]. However, all the deformation-aware methods neglect the keypoint detection phase, limiting their applicability in challenging deformations. Although the problems of keypoint detection and description can be treated separately, recent works that jointly perform detection and description of features [4, 26] indicate an entanglement of the two tasks since the keypoint detection can impact the performance of the descriptor. The descriptor for its turn can be used to determine reliable points optimized for specific goals. In this work, we propose a new method for jointly learning keypoints and descriptors robust to deformations, viewpoint, and illumination changes. We show that the detection phase is critical to obtain robust matching under deformations. Fig. 1 depicts an image pair with challenging deformations, where our method can extract reliable keypoints and match them correctly, significantly increasing matching scores compared to the recent state-of-the-art approach DISK [37].

Contributions. **(1)** Our first contribution is a new end-to-end method called DALF (Deformation-Aware Local Features), which jointly learns to detect keypoints and extract descriptors with a mutual assistance strategy to handle significant non-rigid deformations. Our method boosts the state-of-the-art in this type of feature matching by 8% using only synthetic warps as supervision, showing strong generalization capabilities. We leverage a reinforcement learning algorithm for unified training, combined with spatial transformers that capture deformations by learning context priors affecting the image; **(2)** Second, we introduce a feature fusion approach, a major difference from previous methods that allows the model to tackle challenging deformations with complementary features (with distinctiveness and invariance properties) obtained from both the backbone and the spatial transformer module. This approach is shown beneficial with substantial performance improvements compared to the non-fused features; **(3)** Finally, we demonstrate state-of-the-art results in non-rigid local feature applications for deformable object retrieval and non-rigid 3D surface registration. We also will make the code and both applications publicly available to the community.

2. Related work

Keypoint detection. Traditional image keypoint detection methods seek to extract repeatable regions in images, *i.e.*, localized points that are stable under different viewing conditions. The classic Harris detector [10] employs image derivatives that are used to compute cornerness scores, while one the most used handcrafted detectors SIFT [14], for instance, detects blobs using the Difference of Gaussians. Key.Net [13] showed that it is possible to improve keypoint detection by combining handcrafted filters and learned filters. A recent trend for learning keypoint detection is to couple description and detection in the same pipeline [7, 26, 34, 42], since it is advantageous performance-wise to solve both tasks simultaneously in terms of computation and matching accuracy. In the same direction, our proposed method has a backbone that computes both keypoints and descriptors but at the same time employs a deformation-aware module.

Description of local patches. Until recently, detection and description were treated separately. While some works focused on both problems, such as the seminal works of SIFT [14] and ORB [28], the detection and description were decoupled. SIFT and ORB descriptors employ a handcrafted gradient analysis to extract descriptors with scale and rotation invariance. Recent description approaches based on CNNs [8, 15, 19, 34, 39] consume a local patch assuming a pre-defined keypoint detector. These methods achieved state-of-the-art performance using SIFT keypoints in the image matching benchmark [12]. The networks are trained using metric learning [9, 39]. As patch-based methods rely on a pre-defined keypoint detector that may produce keypoints in unreliable or ambiguous regions, noise can be easily introduced because detection and description steps are decoupled. Unlike patch-based methods, our network is trained to produce reliable descriptions and keypoints optimized for non-rigid correspondence. Our description is also enhanced with a fusion strategy that combines complementary features into a single learned feature representation.

Joint detection and description. DELF [22] and DELG [4] works demonstrated that coupling the detection and description phases using an attentive mechanism for keypoint selection based on higher-level image semantics can substantially improve retrieval performance. Local feature extraction has been shifting towards learning both detection and description of local features jointly [7, 16, 26, 37, 42]. Most methods follow a similar architecture, adopting a fully convolutional network (FCN) layout to produce a dense feature map, where most of the differences between the methods reside in the training scheme and loss design. The most recent describe-and-detect approaches are currently state-of-the-art on standard benchmarks [2, 12]. Differently from existing methods, we design the architecture and training to explicitly handle deformations in a coupled detection and description phase, devising a carefully tailored warper network.

Deformation-aware methods. One of the first proposed image descriptors designed for deformable surfaces is DaLI [30]. DaLI interprets image patches as a local 3D surface, and computes the Scale-Invariant Heat Kernel Signatures [3] of the 3D surface to encode features robust to non-rigid deformations and illumination changes. Despite achieving improved matching performance compared to contemporary works, DaLI suffers from high computational and storage requirements. Similar to DaLI, GeoBit [21] and GeoPatch [24] descriptors brought ideas from computational geometry to computer vision, leveraging RGB-D images to extract visual features that are geodesic-aware. However, these geodesic-aware methods require RGB-D images and are sensitive to noise, significantly restricting their applicability. To remove the need for depth images for estimating geodesic patches, the DEAL descriptor [25] implicitly handles deformations from monocular images through a non-rigid warper module. The main shortcoming of DEAL is its dependency on existing keypoint detectors, which compro-

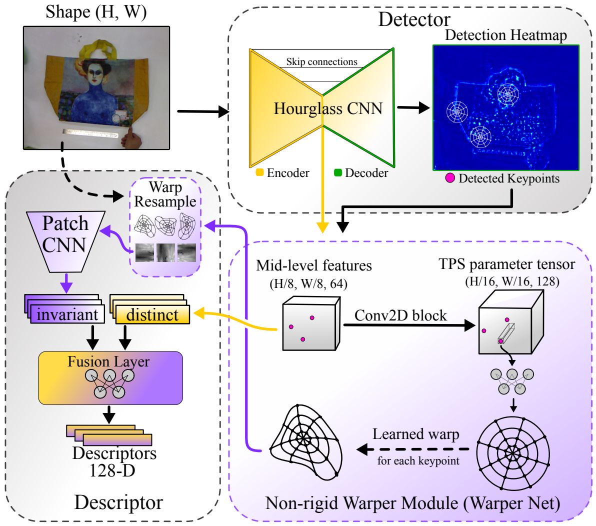

Figure 2. **DALF architecture**. Our architecture jointly optimizes non-rigid keypoint detection and description, and explicitly models local deformations for descriptor extraction during training. An hourglass CNN computes a dense heat map providing specialized keypoints that are used by the Warper Net to extract deformation-aware matches. A feature fusion layer balances the trade-off between invariance and distinctiveness in the final descriptors.

mises the descriptor performance due to the lack of equivariance on keypoints locations from most existing detectors in the presence of deformation changes. In contrast, our method learns detection and description in the same framework, achieving significant performance gains.

3. Methodology

DALF jointly learns to detect and describe points robust to non-rigid deformations, in addition to perspective and illumination changes. Both detector and descriptor are trained with a cooperative scheme aiming at the invariance of feature representations. Specifically, the keypoint detector is trained using policy gradient, seeking to increase the probability of detections that are both repeatable and reliable; Concurrently, the descriptor extractor learns to undeform and extract discriminative and invariant features from local regions. The model is only trained on synthetic warps, *i.e.*, it does not require expensive human annotation nor pseudo-ground-truth that may contain errors and bias, such as the output of an SfM pipeline that is used in several works [7, 16, 26, 29, 37]. Fig. 2 outlines the proposed method.

3.1. Keypoint detector

The keypoint detection architecture uses a backbone hourglass CNN network $\mathbf{f}(\cdot)$, similar to a U-net [27]. This network enables computing a keypoint heat map in the original image resolution efficiently, while also producing mid-level feature representations that are useful to describe the keypoints. We employ three downsampling blocks for the encoder, and three upsampling blocks for the decoder, with skip connections, each having two convolutional layers composed of a 2D convolution followed by ReLU and batch normalization. Let $I \in \mathbb{R}^{h \times w \times c}$ be the input image of size $h \times w$ and c channels, $\mathbf{f}(I)$ outputs two feature maps: mid-level representations $\mathbf{X} \in \mathbb{R}^{h/8 \times w/8 \times d}$ and detection heatmap $\mathbf{H} \in \mathbb{R}^{h \times w}$, where d is the number of features.

Keypoint detection in deforming images. An effective detector must output heatmaps $\mathbf{H} \in \mathbb{R}^{h \times w}$ with high responses in regions that can be matched well in non-rigid scenes containing view and illumination changes. Thus, during the training of the detection branch, we optimize $\mathbf{H}$ using a strategy similar to DISK [37], but applying only the probabilistic framework to learn the detection heatmap. A key difference compared to DISK is that we enforce the reliability of the detected keypoints by penalizing wrong matches even if the keypoints are repeatable. A probabilistic approach has several advantages as dealing with the inhering discreteness of sparse keypoint detection, and a simpler and more intuitive loss can be used for better convergence and regularization of the detection heatmap, in contrast to works that require elaborated handcrafted losses [7, 16, 26].

We seek to obtain high responses in confident regions not only for detection but also matching. To solve this problem with policy gradient, we divide the heatmap into a 2D grid of cells (Detection Heatmap in Fig. 3) and consider a set of actions that the network agent can make to select keypoints. Each cell $\mathbf{c}_i$ has $m \times n$ pixels, where the network can learn the probability of detecting a keypoint within each cell. Given an image pair (A, B) of the same scene under different photometric and geometric transformations, and the ground-truth flow-field relating the two images, for each cell $\mathbf{c}_i \in \mathbf{H}$, we consider a probability distribution over the cell $\mathbf{c} \in \mathbb{R}^{m \times n}$. Each logit value within the cell has a probability of being a keypoint. The probability mass function $\mathbf{p}_{\mathbf{c}_i}$ over the cell $\mathbf{c}_i$ is computed by applying the Softmax function.

Therefore, to train the detection branch, we employ the Reinforce algorithm [32]. During the forward pass of the network, we randomly sample an individual keypoint within each cell $\mathbf{c}_i$ according to the probability mass function $\mathbf{p}_{\mathbf{c}_i}$ alongside the keypoint's spatial coordinates, its probability p_s^i and its logit l_s^i. Note that each cell can have exactly one keypoint; however, in practice, it is common that low texture and ambiguous regions result in low-quality keypoints that cannot be reliably matched or detected in other images. For that reason, we accept a keypoint proposal from a cell with probability $\sigma(l_s^i)$, where σ is the sigmoid activation. This way, the network can learn to filter out unreliable keypoint proposals during training. The final probabilities of detection for the image I is given by the set $P_I = \{\sigma(l_s^i) \cdot p_s^i\}, \forall \mathbf{c}_i \in \mathbf{H}$, such that $\sigma(l_s^i) > 0.5$ (we only sampled keypoints that has positive values in the heatmap). Since we want the

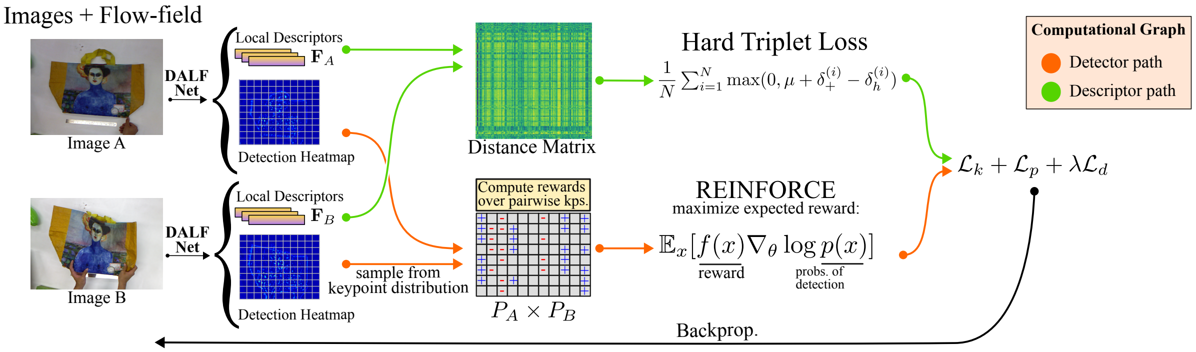

Figure 3. **Training strategy to learn to detect and describe keypoints aware of deformations.** DALF network is used to produce a detection heatmap and a set of local features for each image. In the detector path, the heatmaps are optimized via the REINFORCE algorithm considering keypoint repeatability under deformations. In the descriptor path, feature space is learned via the hard triplet loss. A siamese setup using image pairs is employed to optimize the network. Notice that we penalize keypoints that cannot be described accurately during the second training stage; thus, the keypoints and combined descriptors are optimized jointly to be robust to deformations.

keypoints to be repeatable, we reward points that can be detected in both images A and B. Thus, given the pixel coordinate $\mathbf{p}_A^j \in \mathbb{R}^2$ of detected point j on image A, we define the reward function $R(.)$ as follows:

$$R(\mathbf{p}_A^j) = \begin{cases} 1 & \text{if } \exists \mathbf{p}_B^{(.)} \text{ s.t. } \|T(\mathbf{p}_A^j) - \mathbf{p}_B^{(.)}\| < \tau, \\ 0 & \text{otherwise}, \end{cases} \tag{1}$$

where $T(.)$ transforms pixels coordinates of image A to image B according to the ground-truth flow-field, and τ is a pixel threshold to determine if the detected keypoint in A has a correspondence in image B.

Once we have the set of probabilities P_A and P_B, we obtain the gradient of the parameter vector θ with respect to the expected rewards over all pairwise keypoints $\mathcal{K} = P_A \times P_B$, where $\times$ denotes the cartesian product (see Fig. 3). The gradient ascent is used to maximize the expected sum of rewards:

$$\nabla_\theta \mathbb{E}_{\mathcal{K}}[R(.)] = \sum_{(x,y)\in\mathcal{K}} \nabla_\theta(\log p(x;\theta) + \log p(y;\theta))R(.), \tag{2}$$

where $p(.;\theta)$ denotes the probability of taking that action according to the network parametrized by θ. The variables x and y are the probabilities of detection from a pairwise combination of keypoints.

During the invariant feature learning stage, after 70% of the training progress, we zero out the reward of the keypoints if their descriptors are unreliable. Details about the penalization term for the keypoints are described in Sec. 3.5.

3.2. Keypoint descriptor

We observed that mid-level features extracted from the hourglass encoder do not explicitly model invariance to any kind of deformations, but tend to be highly distinctive on small to moderate photometric and geometric changes, such as varying illumination, and planar warps. Therefore, it is advantageous to supervise the mid-level features since we obtain informative descriptors with no additional cost at inference. For that, during the first training stage, we bilinearly interpolate the feature maps $\mathbf{X}$ at the detected keypoint positions to obtain a feature vector $\mathbf{f}_d \in \mathbb{R}^D$ for each keypoint coordinate. Let $\mathbf{F}_A \in \mathbb{R}^{N\times D}$ and $\mathbf{F}_B \in \mathbb{R}^{N\times D}$ be matrices of N L2-normalized feature vectors of corresponding descriptors $\mathbf{f}_d$ extracted by the hourglass decoder at keypoint positions, from images A and B, and $\mathbf{D}_{N\times N} = \sqrt{2 - 2\mathbf{F}_A\mathbf{F}_B^T}$ the distance matrix. To optimize the descriptors' embedding space, we employ the hard mining strategy [19] in the matrix $\mathbf{D}$ and minimize the margin ranking loss:

$$\mathcal{L}_d\left(\delta_+^{(.)}, \delta_h^{(.)}\right) = \frac{1}{N}\sum_{i=1}^{N}\max(0, \mu + \delta_+^{(i)} - \delta_h^{(i)}), \tag{3}$$

where μ is the margin, $\delta_+ = \|\mathbf{F}(p) - \mathbf{F}(p')\|_2$ is the distance between the corresponding features, and $\delta_h = \|\mathbf{F}(p) - \mathbf{F}(h)\|_2$ is the distance to the hardest negative[1] sample in the batch.

3.3. Non-rigid warper module

CNNs' translation equivariance property makes local descriptors invariant to image translation, and multi-scale strategies increase the robustness of description extraction to in-plane scale changes [7, 26, 37]. However, when non-rigid deformations arise, the local texture can significantly change in appearance, introducing matching ambiguities. DEAL [25] demonstrates that thin-plate-splines (TPS), coupled with spatial transformers, can be used to model deformations for the task of local feature description. Inspired by DEAL, we adopt a TPS deformation to learn local invariance to non-rigid transformations affecting the patches.

[1] The hardest negative example h_i^j for each row j of $\mathbf{D}_{N\times N}$ is computed as min $D_j, i = j = \{1, ..., N\}$ s.t. $i \neq j$.

Spatial transformer network. We use the mid-level features from the backbone network to learn the parameters of the TPS with little additional overhead. The TPS parameter tensor $\mathbf{M}_\theta \in \mathbb{R}^{h/16 \times w/16 \times 2d}$ contains an intermediate representation useful to estimate a local non-rigid transformation for a keypoint. To obtain the parameter vector used in the TPS equation, first, we bilinearly interpolate a feature vector from $\mathbf{M}_\theta$ at the spatial position of the keypoint, obtaining an intermediate parameter vector $\in \mathbb{R}^{2d}$. Then, an MLP is used to estimate the parameter vector μ_θ that is used in the TPS transformation. The parameter vector μ_θ encodes the affine matrix $\mathbf{A} \in \mathbb{R}^{2\times 3}$, and the non-rigid components $\mathbf{w}_k \in \mathbb{R}^2$ separately representing offsets from the affine component. Given a homogeneous 2D point $\mathbf{q} \in \mathbb{R}^3$, weight coefficients and control points $\mathbf{c}_k, \in \mathbb{R}^2$, we use the parameters contained in μ_θ to apply the TPS transformation to a fixed polar grid centered at the keypoint:

$$\mathbf{p} = \mathbf{A}\mathbf{q} + \sum_{k=1}^{n_c} \rho(\|\mathbf{q} - \mathbf{c}_k\|^2)\mathbf{w}_k, \tag{4}$$

where n_c is the number of control points, $\mathbf{q}$ is a normalized spatial image coordinate from the fixed polar grid around the keypoint, and $\mathbf{p}$ is its transformed coordinate. Fig. 2 (Warper Net) shows the patch warping and sampling step. Since we are using the TPS Radial Basis Function, $\rho = r^2 \log r$ is used. After the polar grid is transformed, a differentiable bilinear sampler [11] is used to obtain the transformed image patch that is used by a CNN similar to L2-Net architecture [34] to compute the invariant feature vector, supervised by the margin ranking loss (Eq. (3)). In our implementation, a major difference from the original L2-Net is that in the last convolutional block, we add an average pooling in the axis respective to the angular axis in the polar patch, attaining full rotation invariance.

3.4. Feature fusion layer

Distinctiveness and invariance are two desired attributes of a local feature descriptor. While invariance is vital for tasks that handle large appearance changes, such as rotation and scale, it usually implies distinctiveness loss [38]. By considering two complementary features, the distinctive ones coming from the backbone network with a larger receptive field but more sensitive to strong geometric transformations, and invariant features coming from the warper module that are robust to deformation and rotation by design, we propose to incorporate both information by a feature fusion step.

The fusion is performed by an attention-based MLP that predicts weight coefficients. The two descriptor vectors are first concatenated and forwarded to the Fusion Layer as depicted in Fig. 2. Then, the concatenated descriptors are weighted by the predicted attention weights and L2-normalized to produce the final descriptor. During training, we optimize each descriptor loss individually and the loss of the fused descriptors simultaneously to enforce the network to learn how to fuse the feature vectors to achieve a better feature representation. In the experiments, we demonstrate that combining the features allows the final descriptor to handle strong image transformations while maintaining its distinctiveness.

3.5. Training strategy and model optimization

Stage-wise training. During experiments, we observed that training the network end-to-end in a single phase causes the model to focus on the invariant features and ignores the distinctive features coming from the backbone, even when re-weighting the loss terms. To solve the issue, we perform a two-stage training. During the first training stage, we only train the backbone network. The backbone features have a larger receptive field and higher-level semantics compared to the Warper Net features but with less invariance to rotation and low-level deformations. In the second training phase, the Decoder, Warper Net and Fusion Layer are optimized, where the final feature representation is optimized considering both representations through the fusion step. Moreover, the decoder of the network is further refined and encouraged to detect keypoints that are optimal for the fused descriptors.

Final loss. For the detection branch, we define the keypoint loss as $\mathcal{L}_k = -\mathbb{E}_{\mathcal{K}}[R(.)]$ and add a regularization term for all the detected keypoints during training $\mathcal{L}_p = -\sum_x \log p(x) \cdot c$, where c is a small negative constant to discourage the network from detecting low-quality points. We employ the margin ranking loss described in Sec. 3.2 for all the descriptor vectors computed by our network. The final loss is then computed as $\mathcal{L} = \mathcal{L}_k + \mathcal{L}_p + \lambda\mathcal{L}_d$, where λ is a weight term to balance the magnitude between the triplet loss and the policy-gradient losses.

4. Experiments

Training and implementation details. We developed a carefully designed synthetic data generation pipeline to create plausible non-rigid deformations of surfaces to supervise the training. We perform photometric and geometric changes to real images obtained from large-scale Structure-from-Motion datasets [41]. We use only the raw images and do not use any information about correspondences or annotated labels. During the training, we add random photometric changes, random homographic projection, and random TPS warps to obtain ground-truth dense flows between image pairs depicting the same surface. The training starts with easier samples and progressively becomes more difficult, achieving the hardest difficulty at 60% of training iterations. In the experiments, we use the following hyperparameter values: $\mu = 0.5$ in the triplet loss; pixel threshold $\tau = 1.5$; $\lambda = 0.005$ to balance the loss terms; keypoint penalization

$c = -7e^{-5}$; cell size $m = n = 8$ pixels; and the number of control points $n_c = 64$. As detailed in Sec. 3.5, we perform a two-stage training. Gradient accumulation was used for four forward passes before updating the weights. We trained the network for 80,000 iterations in the first stage and 100,000 iterations in the second stage. During inference, we use a non-maximal suppression of size 3×3 pixels in order to extract the keypoint coordinates from $\mathbf{H}$. Our network is implemented on PyTorch, has about $1M$ trainable parameters, and takes 48 hours to train on a GeForce GTX Titan X GPU.

Baselines and evaluation metrics. We compare our method with several patch-based descriptors [1, 8, 20, 28, 35, 36, 39], using the same set of SIFT [14] keypoints following the protocol of the image matching benchmark [12]. We also perform tests with a detector suitable for non-rigid correspondence [17] coupled with the deformation-aware descriptor DEAL [25]. Finally, we also include in the comparison the state-of-the-art detect-and-describe methods [6, 7, 16, 23, 26, 37, 42]. For each evaluated method, we detect the top 2,048 keypoints and match the descriptors using nearest neighbor search. In addition to the standard comparison, we include as the gold standard for image matching SuperPoint [6] with the SuperGlue [29] matcher, which holds the state-of-the-art for stereo and multi-view camera registration [12] assuming rigid scenes. As shown in Tab. 1, the methods are divided into three categories: (i) methods that only require RGB input (*RGB*) in contrast to methods that require additional information such as depth; (ii) Detect & Describe (*D&D*) methods that provide both detection and description jointly within a single pipeline; and (iii) Deformation-Aware (*D-A*) methods, which take into account deformation when computing the descriptors. Notice that a method may fulfill multiple categories simultaneously.

We used the Matching Scores (MS) [18] to evaluate the matching performance of both the detected keypoints and descriptors. Given a ground-truth transformation and a threshold in pixels, we compute the set of correct correspondences $\mathbf{S}_{gt}$ and obtain the score for an image pair (i, j) as $MS = |\mathbf{S}_{gt}| / \min(|keypoints_i|, |keypoints_j|)$. In addition, the mean matching accuracy (MMA) is also reported, which focuses on the accuracy of the descriptors to match the keypoints that were successfully detected on both images under the threshold denoted as the set $\mathbf{K}_{gt}$, and is computed as $MMA = |\mathbf{S}_{gt}| / |\mathbf{K}_{gt}|$. Additional results regarding repeatability of keypoints can be found in the supplementary material. To conduct the evaluation, we adopt two existing datasets of deformable objects [24, 40].

4.1. Real-world benchmarking

Comparison with the state-of-the-art. Tab. 1 shows the MS and MMA scores achieved by all compared methods. DALF outperformed all descriptors on average in both MS and MMA metrics, including the methods that use additional depth information to extract deformation-invariant features, improving the state-of-the-art in 8% p.p. in matching scores. Moreover, our method shows promising generalization properties to real deformations. DISK achieved the second-best results in MS, but the MMA indicates that its descriptors are more sensitive to non-rigid deformations. DEAL displays the second-best results in MMA thanks to its deformation-aware module but has poor performance on the MS score. It is noteworthy that DEAL relies on SIFT keypoints, which are not designed for non-rigid transformations.

SuperGlue, which is comprised of SuperPoint descriptors matched with a graph neural network, showed good performance, but the drop in the scores compared to the top methods is noticeable as SuperPoint and SuperGlue do not explicitly model scene deformations. We emphasize that our method can be easily coupled and trained with a learned matcher such as SuperGlue. We report the SuperGlue results using the outdoor pretrained weights since we observed that the outdoor weights peformed better in all datasets. All the other methods achieve significantly worse scores due to their inability to cope with stronger deformations alongside illumination and affine transformations.

Rotation and scale robustness. Aside from deformations, in-plane rotations and scale changes are two important geometric transformations. Thus, we conduct a study using the Simulation sequences from [24] containing challenging rotation and scale changes. Fig. 4 clearly indicates that our method holds the best invariance to image in-plane rotations in addition to deformation changes compared to the five stronger competitors. Our technique also displays considerable robustness to scale changes, outperforming SuperPoint and providing a similar level of robustness of SuperGlue.

Time efficiency. DALF is one of the most time efficient methods among the joint detection and description architectures. While our method runs at 9 FPS, DISK runs at 5 FPS and R2D2 at 2 FPS in an NVIDIA GeForce RTX 3060 GPU to extract 2,048 keypoints from 1024×768 images.

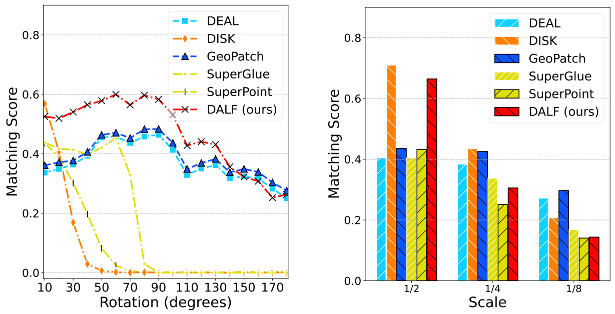

Figure 4. **Invariance to rotation & scale.** We evaluate the matching performance of the methods under rotation and scale changes between image pairs from the *Simulation* dataset. The objects are simultaneously deforming, rotating and scaling in image space.

Table 1. **Performances using top** 2,048 **keypoints**. The *RGB* methods only require color images. *D&D* methods perform joint detection and description. The deformation-aware methods are shown as *D-A*. The mark * indicates the use of a learned matcher. Best in bold and second-best underlined. The mean was calculated with full-precision values by averaging the scores of all 833 image pairs before rounding.

RGB	*D&D*	*D-A*	**Method**	**Datasets:** 833 **pairs total** – MS / MMA @ 3 pixels ↑				**Mean**
				Kinect 1 [24]	*Kinect 2* [24]	*DeSurT* [40]	*Simulation* [24]	
			BRAND [20]	0.17 / 0.34	0.22 / 0.49	0.14 / 0.33	0.04 / 0.09	0.16 / 0.34
✓			ORB [28]	0.19 / 0.38	0.25 / 0.55	0.18 / 0.40	0.14 / 0.30	0.20 / 0.43
✓			DAISY [36]	0.23 / 0.47	0.29 / 0.62	0.16 / 0.37	0.19 / 0.39	0.22 / 0.48
✓			FREAK [1]	0.24 / 0.49	0.33 / 0.72	0.16 / 0.38	0.15 / 0.31	0.23 / 0.51
✓			TFeat [39]	0.25 / 0.50	0.28 / 0.61	0.21 / 0.48	0.29 / 0.63	0.26 / 0.56
✓			Log-Polar [8]	0.28 / 0.58	0.30 / 0.65	0.23 / 0.54	0.22 / 0.49	0.26 / 0.57
✓			SOSNet [35]	0.17 / 0.34	0.25 / 0.55	0.17 / 0.38	0.26 / 0.57	0.22 / 0.47
✓	✓		LF-Net [23]	0.44 / 0.40	0.51 / 0.43	0.28 / 0.77	0.21 / 0.74	0.36 / 0.59
✓	✓		LIFT [42]	0.09 / 0.57	0.16 / 0.65	0.08 / 0.52	0.13 / 0.73	0.12 / 0.62
✓	✓		D2-Net [7]	0.20 / 0.50	0.23 / 0.82	0.14 / 0.47	0.11 / 0.30	0.17 / 0.57
✓	✓		SuperPoint [6]	0.45 / 0.74	0.54 / 0.85	0.39 / 0.68	0.18 / 0.34	0.41 / 0.69
✓	✓		R2D2 [26]	0.17 / 0.36	0.25 / 0.59	0.14 / 0.32	0.06 / 0.16	0.17 / 0.39
✓	✓		ASLFeat [16]	0.31 / 0.58	0.39 / 0.69	0.28 / 0.53	0.19 / 0.35	0.31 / 0.56
✓	✓		DISK [37]	0.53 / 0.76	0.52 / 0.81	0.44 / 0.61	0.26 / 0.34	0.45 / 0.66
✓	✓		SuperGlue* [29]	0.40 / 0.66	**0.62 / 0.99**	0.39 / 0.68	0.23 / 0.43	0.44 / 0.74
		✓	GeoBit [21]	0.31 / 0.65	0.35 / 0.77	0.20 / 0.47	0.32 / 0.71	0.30 / 0.66
		✓	GeoPatch [24]	0.32 / 0.66	0.35 / 0.80	0.26 / 0.60	0.39 / **0.86**	0.33 / 0.73
✓		✓	DaLI [30]	0.25 / 0.51	0.35 / 0.76	0.21 / 0.48	0.10 / 0.22	0.25 / 0.54
✓		✓	SIFT + DEAL [25]	0.33 / 0.68	0.38 / 0.85	0.27 / 0.63	0.36 / 0.80	0.34 / 0.75
✓		✓	Det. [17] + DEAL	0.44 / 0.74	0.49 / 0.82	0.33 / 0.64	0.31 / 0.74	0.40 / 0.74
✓	✓	✓	DALF (ours)	**0.54 / 0.82**	**0.62** / 0.90	**0.49 / 0.73**	**0.42** / 0.69	**0.53 / 0.80**

Table 2. **Ablation.** Performance of our method when considering different network components and training strategies.

Distinct	**Invariant**	**2-Stage**	**Attn.**	↑ MS / MMA
✓		-	-	0.48 / 0.72
	✓	-	-	0.51 / 0.78
✓	✓			0.53 / 0.79
✓	✓	✓		0.53 / 0.78
✓	✓	✓	✓	0.53 / 0.80

4.2. Ablation study

Our ablation study comprises five different configurations of our method: (i) using the U-net backbone only without the non-rigid warper module, which is similar to DISK excepting the descriptor loss term; (ii) computing the descriptors using the non-rigid warper module only; (iii) fusing the invariant and distinct features from the non-rigid warper and backbone respectively; (iv) perform a stage-wise training where the backbone is optimized first and the non-rigid warper second, and finally (v) we perform stage-wise training with an additional attention layer to fuse the invariant and distinctive descriptors instead of simple concatenation.

From Tab. 2, we can observe that the non-rigid warper contributes significantly to achieving more accurate matches when compared to using a convolutional backbone alone. Furthermore, by fusing the features, it is possible to obtain an improved descriptor that is both invariant and distinct with complementary properties. The two-stage training provides similar matching scores and slightly reduced mean accuracy compared to end-to-end training. Still, according to a more detailed analysis available in the supplementary material, we observed that it is beneficial to perform stage-wise training. The invariant part tends to dominate the distinctive part during training, rendering the distinct part less useful in practice, which is not desired for applications needing more distinct features, such as image retrieval, and datasets without significant deformations. Finally, we test if an attention-based fusion layer can deliver better results than concatenating the descriptors in the fusion step. According to the results, it is possible to slightly increase the accuracy even further with a negligible cost in computation. Thus, we choose the model with the stage-wise training as the final architecture.

Limitations. Although our network can improve overall scores by learning keypoints and deformation-aware features, estimating the deformation parameters from a single image is an ambiguous problem. Therefore, physical deformations may make textures similar for different objects, harming our method's performance. Nevertheless, the learned de-

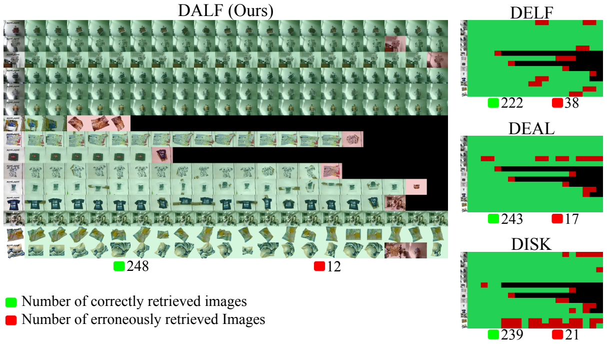
DALF (Ours)
DELF
DEAL
DISK
🟩248
🟥12
🟩222
🟥38
🟩243
🟥17
🟩239
🟥21
🟩 Number of correctly retrieved images
🟥 Number of erroneously retrieved Images

Figure 5. **Deformable object retrieval.** Our method has the best result in retrieving images of deformed objects. The first column shows the image queries, and the rows show the results from different queries. Green images correspond to the same object as the query, and red images do not correspond. Black squares imply no more objects are available for that query.

formations demonstrate good generalization properties to real deformations due to the powerful combination of the keypoint extractor, warper module, and feature fusion steps.

4.3. Applications

To further show the potential usage of our detection and description approach, we performed an evaluation in the two complementary tasks of object retrieval and 3D registration.

Deformable object retrieval. We consider a database that contains images from various deformed objects. Each object appears multiple times with different deformations. The top K images from the database corresponding to an image query are retrieved. To evaluate the methods, we use the retrieval accuracy for different K values. K-Nearest Neighbors is used in conjunction with Bag-of-Visual-Worlds approach over the descriptors as the retrieval engine. We compare our method against the state-of-the-art description methods that demonstrated the top performances in Tab. 1, in addition to DELF [33], a state-of-the-art descriptor designed and trained specifically for image retrieval. We calculated the normalized area under the curve of each method for $K = \{1, ..., 20\}$. DALF achieved the most accurate retrieval capabilities at 99.49%, while DELF, DEAL, SuperPoint, and DISK achieved 98.57%, 98.34%, 97.92%, and 96.12%, respectively. Fig. 5 shows some qualitative results[2].

Non-rigid 3D surface registration. In addition to the retrieval application, we also validate the performance of the methods in the challenging real-world task of surface registration. To that end, we employ the as-rigid-as-possible (ARAP) [31] registration to perform the surface alignment with the correspondences obtained from each method. In these experiments, the correspondences are first filtered with an outlier removal approach [5] since the ARAP cannot handle outliers in the registration process. One of the challenges of non-rigid registration is that one cannot fit a global geometric transformation with a minimal sample using RANSAC.

Table 3. **3D surface registration.** The 2D and 3D accuracy is computed under varying thresholds in centimeters for the 3D residuals and in pixels for the 2D residuals. Best in bold.

Method	**2D Accuracy ↑**			**3D Accuracy ↑**		
	@2px	@3px	@5px	@0.5cm	@1.0cm	@1.5cm
DISK	21.3	30.5	41.2	36.3	51.7	58.8
SuperPoint	23.2	34.4	47.4	42.6	60.5	68.4
SuperGlue	34.9	51.0	**68.1**	42.8	64.4	73.6
GeoPatch	28.9	41.5	55.9	41.0	62.3	71.1
DEAL	29.4	42.0	56.2	42.9	64.3	72.8
DALF (ours)	**36.6**	**51.6**	67.3	**46.2**	**66.9**	**74.8**

After the filtering stage, the ARAP is used to align the meshes of the respective image pairs. The 2D error is then computed using the ground-truth TPS transformation provided with the datasets given in pixels. We also estimate the residual 3D error, assuming that the two corresponding surfaces must be perfectly adjusted in the 3D space, as their meshes are known beforehand. Tab. 3 shows the performance of the top methods under different thresholds for the 2D and the 3D errors considering all the datasets used in Tab. 1, where our approach stands out, improving over 3 p.p. in 3D registration accuracy compared to the best current method (SuperGlue) in the tightest threshold of 0.5cm. Visual results of the 3D registration are available in the supplementary material.

5. Conclusions

This paper presents DALF, a method that considers both the detection and description of keypoints under the challenging case of non-rigid geometric transformations. From extensive experiments and two applications using real deformable objects, we draw the following conclusions: (i) standard approaches for image matching deliver subpar results compared to deformation-aware features; (ii) optimizing the keypoint detection stage together with deformation-aware descriptors brings significant performance gains compared to existing deformation-aware methods that rely on affine keypoint detectors; and (iii) the feature fusion component is a simple but effective approach to increase the network expressiveness to deformations while keeping distinctiveness.

Acknowledgements. The authors would like to thank CAPES (#88881.120236/2016-01), CNPq, FAPEMIG, and Google for funding different parts of this work. This work was also granted access to the HPC resources of CNRS IDRIS under the project 2021-AD011013154 and supported by the French Conseil Régional de Bourgogne-Franche-Comté.

[2] For the sake of clarity, we filtered out objects that all methods retrieved all correct images. Full results are available in the supplementary material.

References

[1] Alexandre Alahi, Raphael Ortiz, and Pierre Vandergheynst. Freak: Fast retina keypoint. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 510–517. Ieee, 2012. 6, 7

[2] Vassileios Balntas, Karel Lenc, Andrea Vedaldi, and Krystian Mikolajczyk. HPatches: A benchmark and evaluation of handcrafted and learned local descriptors. In *CVPR*, 2017. 2

[3] Michael M Bronstein and Iasonas Kokkinos. Scale-invariant heat kernel signatures for non-rigid shape recognition. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 1704–1711. IEEE, 2010. 2

[4] Bingyi Cao, Andre Araujo, and Jack Sim. Unifying deep local and global features for image search. In *European Conference on Computer Vision*, pages 726–743. Springer, 2020. 1, 2

[5] Luca Cavalli, Viktor Larsson, Martin Ralf Oswald, Torsten Sattler, and Marc Pollefeys. Adalam: Revisiting handcrafted outlier detection. *arXiv preprint arXiv:2006.04250*, 2020. 8

[6] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 224–236, 2018. 1, 6, 7

[7] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, and Torsten Sattler. D2-net: A trainable cnn for joint description and detection of local features. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pages 8092–8101, 2019. 2, 3, 4, 6, 7

[8] Patrick Ebel, Anastasiia Mishchuk, Kwang Moo Yi, Pascal Fua, and Eduard Trulls. Beyond cartesian representations for local descriptors. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 253–262, 2019. 2, 6, 7

[9] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742. IEEE, 2006. 2

[10] Christopher Harris and Mike Stephens. A combined corner and edge detector. In *Proceedings of the Alvey Vision Conference*, pages 23.1–23.6, 1988. 2

[11] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. *Advances in neural information processing systems*, 28:2017–2025, 2015. 5

[12] Yuhe Jin, Dmytro Mishkin, Anastasiia Mishchuk, Jiri Matas, Pascal Fua, Kwang Moo Yi, and Eduard Trulls. Image matching across wide baselines: From paper to practice. *International Journal of Computer Vision*, pages 1–31, 2020. 2, 6

[13] Axel Barroso Laguna, Edgar Riba, Daniel Ponsa, and Krystian Mikolajczyk. Key. net: Keypoint detection by handcrafted and learned cnn filters. *arXiv preprint arXiv:1904.00889*, 2019. 2

[14] David G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, pages 91–110, 2004. 2, 6

[15] Zixin Luo, Tianwei Shen, Lei Zhou, Siyu Zhu, Runze Zhang, Yao Yao, Tian Fang, and Long Quan. Geodesc: Learning local descriptors by integrating geometry constraints. In *Proceedings of the European conference on computer vision (ECCV)*, pages 168–183, 2018. 2

[16] Zixin Luo, Lei Zhou, Xuyang Bai, Hongkai Chen, Jiahui Zhang, Yao Yao, Shiwei Li, Tian Fang, and Long Quan. Aslfeat: Learning local features of accurate shape and localization. In *CVPR*, pages 6589–6598, 2020. 2, 3, 6, 7

[17] Welerson Melo, Guilherme Potje, Felipe Cadar, Renato Martins, and Erickson R Nascimento. Learning to detect good keypoints to match non-rigid objects in rgb images. In *SIBGRAPI*, volume 1, pages 61–66. IEEE, 2022. 6, 7

[18] Krystian Mikolajczyk, Tinne Tuytelaars, Cordelia Schmid, Andrew Zisserman, Jiri Matas, Frederik Schaffalitzky, Timor Kadir, and Luc Van Gool. A comparison of affine region detectors. *International journal of computer vision*, 65(1-2):43–72, 2005. 6

[19] Anastasiia Mishchuk, Dmytro Mishkin, Filip Radenovic, and Jiri Matas. Working hard to know your neighbor's margins: Local descriptor learning loss. In *Advances in Neural Information Processing Systems*, pages 4826–4837, 2017. 1, 2, 4

[20] Erickson R. Nascimento, Gabriel L. Oliveira, Mario Fernando Montenegro Campos, Antônio Wilson Vieira, and William Robson Schwartz. BRAND: A robust appearance and depth descriptor for RGB-D images. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2012, Vilamoura, Algarve, Portugal, October 7-12, 2012*, pages 1720–1726. IEEE, 2012. 6, 7

[21] Erickson R. Nascimento, Guilherme Potje, Renato Martins, Felipe Cadar, Mario F. M. Campos, and Ruzena Bajcsy. GEOBIT: A geodesic-based binary descriptor invariant to non-rigid deformations for RGB-D images. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10003–10011, 2019. 2, 7

[22] H. Noh, A. Araujo, J. Sim, T. Weyand, and B. Han. Large-scale image retrieval with attentive deep local features. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 3476–3485, 2017. 2

[23] Yuki Ono, Eduard Trulls, Pascal Fua, and Kwang Moo Yi. Lf-net: learning local features from images. In *Advances in neural information processing systems*, pages 6234–6244, 2018. 6, 7

[24] Guilherme Potje, Renato Martins, Felipe Cadar, and Erickson R Nascimento. Learning geodesic-aware local features from rgb-d images. *Computer Vision and Image Understanding*, 219:103409, 2022. 1, 2, 6, 7

[25] Guilherme Potje, Renato Martins, Felipe Chamone, and Erickson Nascimento. Extracting deformation-aware local features by learning to deform. *Advances in Neural Information Processing Systems*, 34:10759–10771, 2021. 1, 2, 4, 6, 7

[26] Jerome Revaud, Cesar De Souza, Martin Humenberger, and Philippe Weinzaepfel. R2D2: Reliable and repeatable detector and descriptor. In *Advances in Neural Information Processing Systems*, volume 32, pages 12405–12415, 2019. 1, 2, 3, 4, 6, 7

[27] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 3

[28] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. ORB: an efficient alternative to SIFT or SURF. In *ICCV*, Barcelona, 2011. 2, 6, 7

[29] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4938–4947, 2020. 3, 6, 7

[30] Edgar Simo-Serra, Carme Torras, and Francesc Moreno-Noguer. DaLI: deformation and light invariant descriptor. *International Journal of Computer Vision*, 115(2), 2015. 1, 2, 7

[31] Olga Sorkine and Marc Alexa. As-rigid-as-possible surface modeling. In *Symposium on Geometry processing*, volume 4, pages 109–116, 2007. 8

[32] Richard S Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. *Advances in neural information processing systems*, 12, 1999. 3

[33] Marvin Teichmann, Andre Araujo, Menglong Zhu, and Jack Sim. Detect-to-retrieve: Efficient regional aggregation for image search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5109–5118, 2019. 8

[34] Yurun Tian, Bin Fan, and Fuchao Wu. L2-net: Deep learning of discriminative patch descriptor in euclidean space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 661–669, 2017. 1, 2, 5

[35] Yurun Tian, Xin Yu, Bin Fan, Fuchao Wu, Huub Heijnen, and Vassileios Balntas. Sosnet: Second order similarity regularization for local descriptor learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11016–11025, 2019. 6, 7

[36] Engin Tola, Vincent Lepetit, and Pascal Fua. Daisy: An efficient dense descriptor applied to wide baseline stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(5):815–830, 2010. 6, 7

[37] Michał Tyszkiewicz, Pascal Fua, and Eduard Trulls. Disk: Learning local features with policy gradient. *Advances in Neural Information Processing Systems*, 33:14254–14265, 2020. 1, 2, 3, 4, 6, 7

[38] Manik Varma and Debajyoti Ray. Learning the discriminative power-invariance trade-off. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8. IEEE, 2007. 5

[39] Daniel Ponsa Vassileios Balntas, Edgar Riba and Krystian Mikolajczyk. Learning local feature descriptors with triplets and shallow convolutional neural networks. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2016. 2, 6, 7

[40] Tao Wang, Haibin Ling, Congyan Lang, Songhe Feng, and Xiaohui Hou. Deformable surface tracking by graph matching. In *IEEE International Conference on Computer Vision*, 2019. 6, 7

[41] Kyle Wilson and Noah Snavely. Robust global translations with 1dsfm. In *European Conference on Computer Vision*, pages 61–75. Springer, 2014. 5

[42] Kwang Moo Yi, Eduard Trulls, Vincent Lepetit, and Pascal Fua. LIFT: Learned invariant feature transform. In *ECCV*, 2016. 1, 2, 6, 7